Final Revision

3rd.Prep – Second Term "2023"

إعــــداد وتــصميـــم/ جـــروب فـريــق أصـــدقـــاء الكمبيوتـــر – أ/يـــاسميـن شعـيب

❀Chapter One: Data Types❀

In VB.net language, there are many Data Types.

✍Some Data Types provided by (Visual Basic.Net) are :

1) Numerical:

a- Integral : (Byte – Long – Integer – Short).

b- Non Integral : (Double – Single – Decimal).

2) Character : (String – Char).

3) Miscellaneous : (Object – Date – Boolean).

▪ Each classification of Data Type has more than one type.

▪ Each Data Type has:

→A storage space in the memory: for example the data type (Integer); when used, it occupies (4 bytes) .

→A range of values (minimum value and maximum value); for example the range of values for the data type (Byte) starts with '0' and ends with '255'.

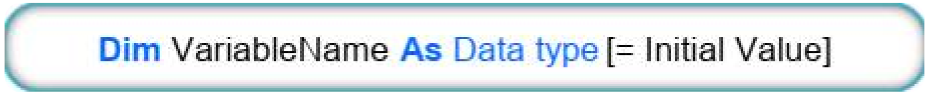
♦ Variables are reserved places in computer memory (RAM) to store values temporarily, these values change during the running of program.

♦ Each Variable has a name and Data type.

♦ These Data Types are created by declaring variables then assigning values to them depending on their Data Types.

✍Declaring Variables:

→ We use "Dim" keyword to declare the variable.



```
Dim VariableName As Data type [= Initial Value]
```

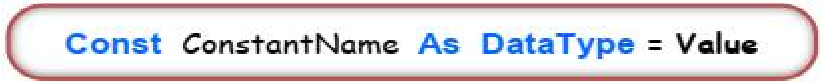
♦ Constants They are places reserved in the (RAM) and, have data types.

♦ You should be declaring the constants and assigning values of them.

♦ These values are a fixed and cannot be changed during program execution.

✍Declaring Constants:

→We use the (Const) keyword to declare constants.



```
Const ConstantName As DataType = Value
```

❀ The double quotes " " are used if the value of variable or constant is a string value.

❀ The hashes # # are used if the value of variable or constant is date or time.

✍Naming Rules Variables & Constants:

1) Variable or Constant names must begin with a letter or underscore (_).

2) Variable or Constant names should not contain symbols or special characters (e.g.: ?, *, ^, -, +, ….. etc.) .

3) Variable or Constant names consist of letters, numbers, and underscores (_).

4) Do not use reserved words (Visual Basic.NET Language Keywords) such as (single, Dim, As).

5) It is preferable that the Variable or Constant name reflects its content.

Assignment Statement:

→It is a statement consists of two sides (right hand side and left hand side) separated by the assignment operator (=).

→It consists of taking the value on the right side of the assignment operator (=) and storing it in the element on the left.

Notes:

→"Me" expresses the current window Form.

→Separates each variable and the other by the concatenation symbol "&".

→The reserved word (vbCrLf) is used to create a new line.

→Use the symbol (_) to write on more than one line if the code line is too long so you can organize and facilitate the process of reading the (Code).

→The programmer can use the command (Rem) in writing remarks that can be referred to within the code, it is not compiled.

Priority rules for Arithmetic operations:

1. Applying the process inside the brackets from the inside to the outside.
2. Applying the exponent.
3. Applying multiplication or division process from left to right,
   wherever comes first.
4. Finally, the Application of the addition or subtraction process
   from left to right, wherever comes first.

Errors:

1- Syntax Error:

This happens, when writing code incorrectly.

→Example: Din x As Single

The variable (X) was declared but there is a mistake in writing the word (Dim)

2- Logical Error:

It happens when we get incorrect results after executing the program because of the wrong formulating arithmetic or logic expressions

→Example:

Write the code => label2.text = Pi + Radius ^2 instead of The code => label2.text = Pi * Radius ^2 to calculate the Area of a Circle.

→ To overcome this type of error, you must review the written code, and test the program with data already validated

3- Runtime Error:

→These errors are discovered when the program is running.

→ It is found in lines of code, where the Assignment Statement is written.

→Example:

when declaring a variable of type Byte and during the program running, a value that is less than or greater than the allowable range is given, i.e. less than (0) or greater than (255) so an error appears during the run, meaning that the value is out of range.

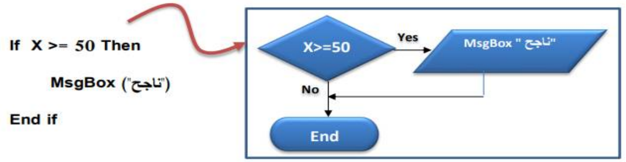
❀Chapter Two: Branching❀

❀Branching Statement using (If…Then):

The syntax of (If…Then) statement

If conditional Expression Then
Code
End if

If the result of conditional expression is "True", the statement after "Then" is executes until "End If", and if the result is "False", executes after "End If"



```
If  X >= 50 Then
      MsgBox ("ناجح")
End  if
```

✍Conditional expression

→ is part of a program code; its result can be (True) or (False) depending on the value of: (a Property or a Variable or another piece of data in the program code).

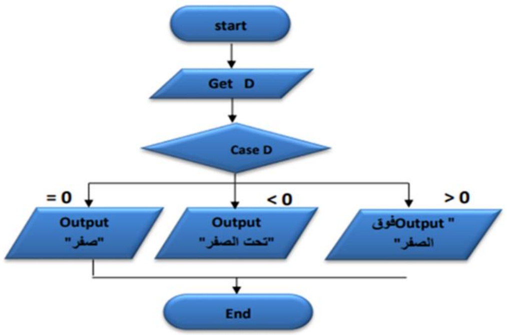
→It consists of three parts:

logical operator preceded by an abstract value.
value of a variable or constant or
result of mathematical expression

❀Branching Statement using (If…Then…Else):

The syntax of (If…Then…Else) statement

If conditional Expression Then
Code → "True"
Else
Code → "False"
End if

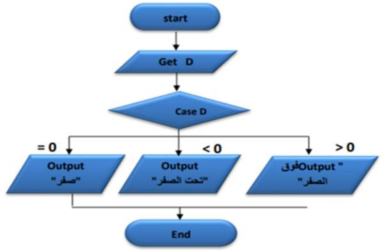


⇨This (If) statement can be written, in one line without writing (End if)

⇨Where (Mod) is a mathematical operator that computes the remainder of a division expression; when the remainder of a division by 2 equals zero, this means there is no remainder; therefore it is an even number.

❀Branching Statement using (Select … Case):

Select …Case *Variable*
Case value1
Code
Case value2
Code
Case value3
Code
Case else
Code
End Select



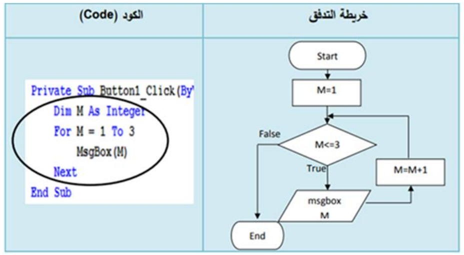
❁Chapter Three: Looping & Procedures❁

❁Looping using (For…Next):

It is one of the limited loop statements used when we want to repeat a code for specific number of time.

→ The syntax of the (For…Next) statement

```
For Variable = Start Value To End Value [Step Add Value]
VB code
Next [Variable]
```



```
Private Sub Button1_Click(By'
    Dim M As Integer
    For M = 1 To 3
        MsgBox(M)
    Next
End Sub
```

- The (For…Next) statement should start with the Keyword (For) and end with the Keyword (Next).
- "Variable" is the name which represents the counter and its type must be numeric (integer or decimal).
- "Start Value" is the start value of the counter or the beginning of repetition is a numeric value.
- "End Value" is the value of the end of the counter and the end of the repetition is also a numeric value.
- "Add Value" is the increment value of the counter or value over the counter until it reaches the end value.
- "Code" is one or more commands to be replicated between the beginning of loop (For) and its end (Next).
- The statements (VB code) are repeated inside the loop until the counter value reaches the End value.
- If the value of the increment is positive 1, it can be dispensed with writing (Step Add Value) as the default value to increase the counter positive 1.

❁Looping using (Do While… Loop):

→ The syntax of the (Do While…..Loop):

```
Do While Conditional Expression
VB code
Loop
```

⇨The (Do While…Loop) is used to repeat a specific code for a several times of an unknown end, but based on a specific condition, so it is useful if you do not know the number of iterations .

⇨The code between the beginning of the loop "Do While" and its end will be implemented as long as the conditional expression is true.

If the condition is not met for any reason, we get out of the iterative loop, and implement the code after the Loop if it exists.

❁Procedure

A set of commands and instructions under a name, can be recalled by that name, to implement them, and create a (Sub) if we have a set of commands that are frequently used in more than one place in the class.

⇨Procedures must be called by their names.

⇨ Calling a procedure causes the program to execute procedure's statements or code.

⇨ Variables and constants, that can be declared either within the scope of the (Event procedures) or within the (Class).

⇨ You can declare the Procedures, this declaration is done only once, but you recall the procedures many times from anywhere in your program.

There are two types of procedures in Visual Basic .NET:

Sub procedures( )اجراء فرعى

Sub procedures do not return a value ( ),لايعود بقيمةwhile

Functions ( )الدالةprocedures.

Functions return a value.

Declaration of Sub Procedures:

You can declare a Sub procedure in a class; if we had a code that will be repeated in more than one place in this Class; as well as for the organization of this code, and so it will be easy to read and understand.

And then modify it if necessary.

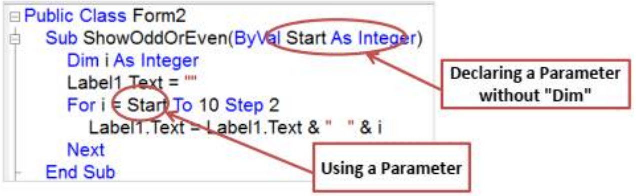
Declaration Syntax:

Sub Name (Parameters)
Code
End Sub

1) "Name" reflects the name of the procedure.
2) "Parameters" reflect the values that were used inside the procedure code that are used on recalling the procedure.
3) "Code" is a set of orders and instructions carried out on recalling the procedure (Sub).

Declaring and using parameters:

To solve the previous problem, the procedure (ShowOddOrEven) must receive the values (1) or (2) on recalling it. this value is used to specify whether the odd numbers will be displayed or the even numbers will be displayed. So we add the variable (Start) that will be called later.



```
Public Class Form2
    Sub ShowOddOrEven(ByVal Start As Integer)
        Dim i As Integer
        Label1.Text = ""
        For i = Start To 10 Step 2
            Label1.Text = Label1.Text & "   " & i
        Next
    End Sub
```

In figure, a sub Procedure of the name (ShowOddOrEven) has been declared and a Parameter named (Start) has been also declared. And used in the code to specify the starting value of the iteration (repetition), accordingly it display.

Declaration of Function Procedures:

Function is a set of commands under a particular name that should express its task. It is applied to Parameters and Returns a value

Declaration Syntax:

Function Function Name (Parameters) As Data Type
Code
Return Value
End Function

Where:

1) "Name" expresses the name of the function.
2) "Datatype" identifies the type of the returned value of the function.

3) "Parameters" represents the parameters that will be used in the code.

4) "Code" is a set of commands and instructions that will be executed on calling the Function.

5) "Value" is the returned value by the function.

⇨ Remember:

1) Variables: We can assign values to Variables; during the declaration and the execution of the Program instructions, *as* well as using these values stored.

2) Constants: We can assign values to Constants; during the declaration only, as well as using these values stored.

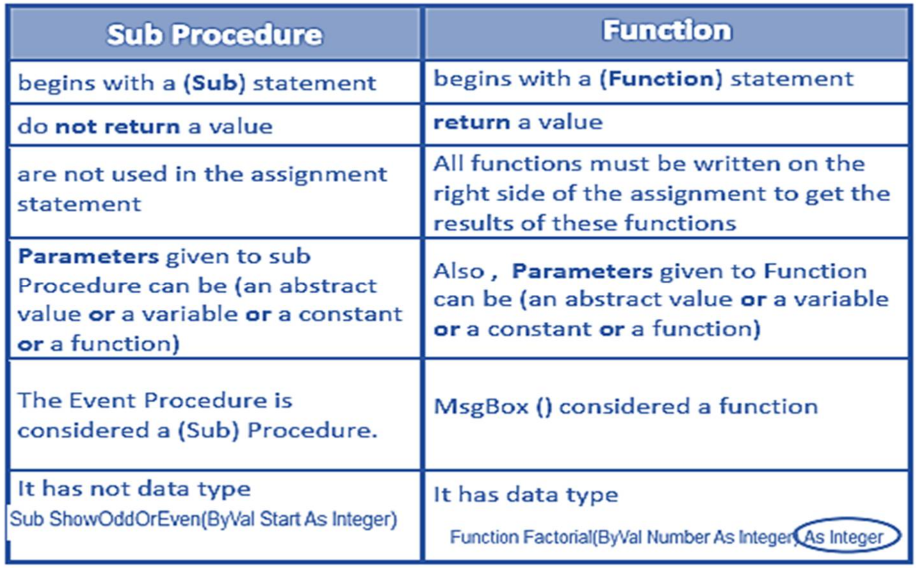
3) Functions: We cannot assign values to Functions, but function can be called and returns a value in the light of the values assigned to the function, as well as using this value stored.

⇨ Remember:

It is preferred when naming Functions; give names related to their functionality. You can:

1. Declare a Function.
2. Determine its Parameters.
3. Specify the Function type.
4. Write Code within this Function.
5. Return a value using the Return statement.

⇨ Remember:



Sub Procedure	Function
begins with a **(Sub)** statement	begins with a **(Function)** statement
do **not return** a value	**return** a value
are not used in the assignment statement	All functions must be written on the right side of the assignment to get the results of these functions
Parameters given to sub Procedure can be (an abstract value **or** a variable **or** a constant **or** a function)	Also , **Parameters** given to Function can be (an abstract value **or** a variable **or** a constant **or** a function)
The Event Procedure is considered a (Sub) Procedure.	MsgBox () considered a function
It has not data type Sub ShowOddOrEven(ByVal Start As Integer)	It has data type Function Factorial(ByVal Number As Integer) As Integer

❀ Chapter Four: Cyber Bullying ❀

⇨ There are many risks that we can be exposed to, including:

⇨ getting wrong information.

⇨ falling prey to some of the aggressors across modes of electronic communication.

⇨ violation of privacy.

⇨ identity theft.

⇨ getting our account stolen (on the social networking sites like Facebook or email).

⇨ Subjecting our system to the risk of infection by viruses or spyware, or software piracy and others. . getting wrong information.

⇨**Cyber bullying** is a deliberate aggressive عدوان متعمد behavior from one person to another through electronic modes of communication.

⇨Firstly: The forms of cyber bullying:

1. Harassment
2. annoyance
3. embarrassment
4. intimidation
5. threat
6. Blackmailing
7. …. Etc.

⇨Secondly: The Electronic Media

Electronic media is a technology used by the electronic aggressor, and they are various including the following:

1. Email.
2. Forums.
3. Instant Message.
4. Facebook.
5. Blogger.

⇨Thirdly: Forms of Cyber Bullying

Forms of cyber bullying include:

1. Anonymity:

It is the use of pseudonyms (aliases) to hide e-aggressor's identity for impunity.

2. Harassment:

It is aggressive messages directed against one or more persons.

3. Cyber stalking:

It is a form of electronic harassment where the aggressor frequently traces and chases a particular person in all electronic media.

4. Flaming:

It is a publication of hostile and vulgar words against one or more through a media and electronic communication.

5. Outing:

It is a dissemination of information about a specific person or more abusively.

6. Exclusion:

It *is* to ignore one or more persons through the electronic media.

7. Cyber threats :

It is an email or e-message carrying a threat and intimidation to one or more persons.

⇨Fourthly: How to protect yourself from Cyber bullying?

By following the safe use of the internet as follows:

1- Don't share your password with anyone.
2-Make a password that is difficult to predict.
3-Don't publish (post) any private data.
4-Avoid deleting Cyber bullying messages.
5- Don't interview anyone you know via the internet
6- Be careful! Don't send any electronic message when you are angry.
7-Inform your parents with what annoy you when you use the internet.
8-The download of software from the internet should be done under the supervision of your teacher or your parents.